

# Manual de Instalación

**INDICE – Manual de instalación**

7	Manual de Instalación .....	393
7.1	Introducción .....	393
7.2	J2SDK .....	393
7.3	Kawa .....	394
7.4	Apache .....	394
7.5	Jakarta-Tomcat.....	396
7.6	Unión Apache con Jakarta-Tomcat.....	397
7.7	Instalación de Java Mail.....	398
7.8	Instalación de RowSet.....	399
7.9	MS Access.....	399
7.10	Instalación de nuestra herramienta.....	400
7.11	Bibliografía .....	401

## 7 Manual de Instalación

---

### 7.1 Introducción

Se describirá como el administrador local deberá de instalar y mantener el software necesario para el correcto funcionamiento de la herramienta. También se explicará qué herramientas software fueron necesarias instalar y configurar para el desarrollo de esta aplicación así como el modo en el que estas fueron instaladas y configuradas.

A continuación se describirán las distintas herramientas utilizadas:

### 7.2 J2SDK



Este es el entorno de desarrollo para Java proporcionado por Sun Microsystems y que utilizaremos para compilar nuestras clases Java. También nos proporciona la herramienta **JavaDoc**, que nos permitirá documentar nuestras clases.

El **J2SDK** lo podremos bajar desde:

<http://java.sun.com/j2se/1.3/download.html>

Una vez descargado el archivo ejecutable **j2sdk-1\_3\_1\_04-win.exe** procederemos a su instalación. Una vez instalado el compilador, tendremos que crear una serie de variables de entorno para acabar la instalación y poder ponernos a trabajar con el entorno de desarrollo.

Para la creación de las variables deberemos de acudir (en Windows XP) a:

**Mi PC > Propiedades > Opciones Avanzadas > Variables de Entorno.**

Una vez aquí deberemos de crear las siguientes dos variables de entorno:

**CLASSPATH = <PATH J2SDK>;**

Así indicaremos al compilador Java donde ha de ir para encontrar los directorios que contienen las clases o librerías de Java (el API de Java). En un futuro tendremos que añadir a la variable **CLASSPATH** aquellos nuevos directorios en los que queremos que el compilador sea accesible a nuevas clases de usuario.

**PATH = <PATH J2SDK>\bin;**

El desarrollo y ejecución de aplicaciones en Java exige que las herramientas para compilar (**javac.exe**) y ejecutar (**java.exe**) se encuentren accesibles. El ordenador, desde una ventana de comandos de MS-DOS, sólo es capaz de ejecutar programas que se encuentre en los directorios indicados en la variable **PATH** del ordenador. Si se desea compilar o ejecutar código en Java en estos casos el directorio donde se encuentran estos programas (**java.exe** y **javac.exe**) deberán encontrarse en el **PATH**.

### 7.3 Kawa



El **J2SDK**, como ya hemos comentado se trata de un entorno de desarrollo muy básico en el que se proporciona básicamente el compilador. En cambio no nos proporciona ningún soporte para programar nuestras clases Java, teniendo que acudir al Bloc de Notas de Windows para implementar nuestras clases. **Kawa**, aun siendo muy básico, nos facilita la implementación de nuestras clases Java, para lo cual nos permite editar código Java de una forma fácil y sencilla.

**Kawa** no posee un compilador Java, para lo cual se ayuda de otros compiladores Java que estén instalados en el sistema, en nuestro caso se ayudará del instalado con el **J2SDK**. De igual modo también puede utilizar otras herramientas, como es la herramienta **JavaDoc**, que se podrá usar para documentar las clases. También puede utilizar la ayuda proporcionada por **J2SDK** sobre el API de Java.

Para indicar a **Kawa** donde debe de acudir para poder utilizar estas dos herramientas y la ayuda deberemos ir a:

#### Menu Kawa > Customize > Options...

Desde aquí podremos indicarle la ubicación exacta de cada una de estas dos herramientas y de las páginas Web que conforman el API de Java.

Podremos acudir a la página de Macromedia para poder bajarnos alguna versión de prueba de **Kawa**.

<http://www.macromedia.com/software/kawa/>

### 7.4 Apache



En primer lugar acudiremos al sitio oficial donde podremos descargarlos gratuitamente este servidor Web, para proceder a continuación a su instalación.

<http://www.apache.org>

En la dirección anterior nos podremos bajar la versión 1.3.23 del **Apache** para plataformas Windows, que es la versión utilizada en este proyecto, en este caso el nombre del archivo es **apache\_1.3.23-win32-x86-no\_scr.msi**.

Para la utilización de apache nuestro sistema debe cumplir un requerimiento, tener instalado el protocolo TCP/IP.

Cuando comience la instalación, se nos preguntará si queremos instalar el apache como un servicio o que arranque manualmente, elegiremos el arranque manual. Seguidamente se ha de elegir entre dos tipos de instalación, la Custom que instala la documentación y/o binarios y la Completa, que instala también la

documentación y los binarios pero que requiere más espacio. Elegiremos la opción Custom y seleccionaremos el directorio en el que queremos que Apache sea instalado, por ejemplo en:

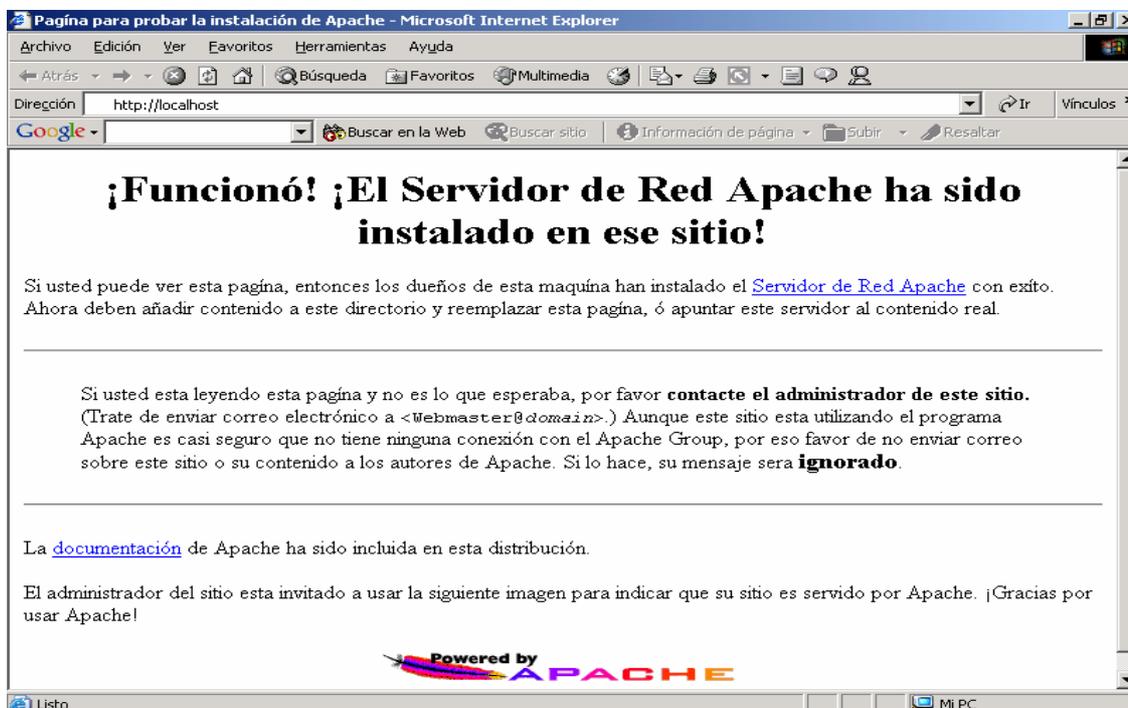
### C:\ Apache

Una vez instalado habrá que reiniciar el ordenador. Para arrancarlo no tenemos más que ir a Inicio > Programas > Apache HTTP Server > Console o bien en la carpeta donde tenemos instalado Apache, ejecutar el archivo **Apache.exe**.

El siguiente paso será comprobar su correcto funcionamiento, para lo cual deberemos de abrir una ventana del explorador y poner en la barra de navegación:

<http://localhost> o <http://nombreequipo>

Si aparece la siguiente pantalla es que la instalación ha sido correcta.



Para parar el servidor podremos ejecutar en una ventana de comandos situados en el directorio donde tengamos el Apache instalado la instrucción: **Apache -k shutdown**. Otra manera es pulsar las teclas Ctrl+C o cerrar la ventana de Apache (esto no es muy conveniente, ya que pueden quedar transacciones sin terminar)

## 7.5 Jakarta-Tomcat



El **Yakarta-Tomcat** es un servidor de páginas Web dinámicas que también puede servir páginas estáticas. El motivo de su instalación junto con el Apache es por que si se hace cooperar ambos podremos servir las páginas Web de una forma más eficiente. Apache sirviendo las páginas Web estáticas y el **Tomcat** las dinámicas.

Vamos a indicar las instrucciones que se han de seguir para poder instalar el **Tomcat 4.0**.

En primer lugar deberemos de acudir al sitio Web donde podremos descargar las fuentes del servidor:

<http://jakarta.apache.org/tomcat/>

Una vez en el sitio Web deberemos de bajarnos el fichero **jakarta-tomcat-4.0.3.zip** y lo descomprimiremos donde queremos que se ubique el servidor, por ejemplo en un directorio llamado:

**C:\JakartaTomcat**

El siguiente paso es la creación de las variables de entorno (acudiremos al mismo sitio que para la creación de las variables del J2SDK)

**Mi PC > Propiedades > Opciones Avanzadas > Variables de Entorno.**

Aquí añadiremos las siguientes variables de entorno:

**TOMCAT\_HOME= <path JakartaTomcat>**

**JAVA\_HOME = <path J2SDK>**

Y editaremos la variable de entorno **CLASSPATH** añadiendo la siguiente ruta al final:

**<path J2SDK>\lib;<path Jakarta Tomcat>\common\lib\Servlet.jar**

El último paso para acabar la instalación es el reiniciar el ordenador para que los cambios surtan efecto. Para hacer funcionar el **Tomcat** como servidor habrá que ejecutar el fichero **startup.bat** que está en la carpeta **bin** de **Jakarta-Tomcat**.

El siguiente paso será comprobar su correcto funcionamiento, para lo cual deberemos de abrir una ventana del explorador y poner en la barra de navegación:

<http://localhost:8080> o <http://nombreservidor:8080>

Si la página que aparece es la siguiente es que **Jakarta-Tomcat** esta correctamente instalado.



Del mismo modo para parar el servidor tendremos que ejecutar el archivo **shutdown.bat** que se encuentra en la carpeta **bin** del **Jakarta-Tomcat**.

## 7.6 Unión Apache con Jakarta-Tomcat

La unión entre **Apache** y **Jakarta-Tomcat** permitirá que ambos servidores trabajen conjuntamente a la hora de servir las páginas Web tanto estáticas como las dinámicas de una forma mucho más eficiente que tener sólo instalado el **Jakarta-Tomcat**.

En el mismo sitio Web donde descargamos el servidor **Apache**:

<http://www.apache.org>

Deberemos de bajarnos el archivo que nos permitirá hacer la conexión entre los dos servidores, **webapp-module-1.0-tc40-windows.zip**. Al descomprimir este fichero vemos que contiene, a parte de los ficheros habituales de licencia y demás, dos ficheros:

**libapr.dll**  
**mod\_webapp.so**

Una vez descomprimidos deberemos de copiar el archivo **libapr.dll** a la siguiente carpeta:

**<PATH Windows>\system32**

El siguiente paso es copiar los dos archivos **libapr.dll** y **mod\_webapp.so** en la carpeta **modules** del **Apache**.

El penúltimo paso es editar el fichero de configuración del **Apache**, **httpd.conf**, donde habrá que ir añadiendo o modificando una serie de líneas:

- 1) Añadiremos las siguientes líneas al final de la sección LoadModule:

```
LoadModule webapp_module
modules/mod_webapp.so
```

- 2) En la sección ClearModuleList añadir:

```
AddModule mod_webapp.c
```

- 3) Añadir la siguientes líneas al final del fichero httpd.conf:

```
WebAppConnection warpConnection warp localhost:8008
WebAppDeploy examples warpConnection/examples/
WebAppInfo /webapp-info
```

Para acabar se debe de arrancar el **Tomcat** primero y después el **Apache** y se probará la siguiente dirección:

<http://nombreservidor> o <http://localhost>

## 7.7 Instalación de Java Mail

La API de **JavaMail**, es un paquete opcional necesario para el envío de correos electrónicos por parte de nuestra aplicación. La instalación se refiere a una Máquina con Windows NT, para otros sistemas operativos el mecanismo de instalación es muy parecido.

El primer paso será bajarnos el paquete JavaMail desde la siguiente dirección:

<http://java.sun.com/products/javamail/index.html>

El paquete viene comprimido en el archivo **javamail-1\_3.zip**, una vez descomprimido en un subdirectorio el paquete, se tendrá que añadir, como en casos anteriores, el paquete en la variable de entorno **CLASSPATH**. Para lo cual basta con añadir al final de la variable la siguiente línea:

**<paht JavaMail>\mail.jar**

Completado este paso, pasaremos a la última tarea de la instalación, que será la descarga y configuración del marco de trabajo de activación de JavaBeans. Esta es una API referida por **JavaMail** y por tanto habrá que instalarlo también en el sistema.

El **Java Activation Framework** puede ser descargado en la dirección:

<http://java.sun.com/products/javabeans/glasgow/jaf.html>

Una vez descargado el archivo **jaf-1\_0\_2.zip** y descomprimido, basta con incluir en la variable **CLASSPATH** el archivo **activation.jar** para lo cual se añadirá la siguiente línea en la variable:

**<path Java Activation Framework>\activation.jar**

## 7.8 Instalación de RowSet

En la versión 2.0 del API JDBC se introduce a la implementación de la interfaz **RowSet**. Este permite entre otras cosas la utilización de la interfaz **CachedRowSet**, que proporciona características muy útiles en el desarrollo de páginas JSP. El objeto **CachedRowSet** proporciona un contenedor para datos tabulares que pueden ser serializados y desplazados, y que pueden ser visualizados sin necesidad de una conexión permanente con la base de datos. Es decir, un objeto **CachedRowSet** puede considerarse como un conjunto de registros de una base de datos que son almacenados fuera de la fuente de datos en memoria. Esta característica resulta extremadamente útil en aplicaciones Web en la cuales el uso de Enterprise JavaBeans resulta excesivo.

Este objeto puede ser considerado como un objeto **ResultSet** desconectado. Implementa la interfaz **RowSet** que soporta el modelo JavaBean y proporciona todos los métodos y características del objeto **ResultSet**, pero no necesita estar continuamente conectado a la base de datos. Esta característica es la que permite que los objetos **ResultSet** sean serializados, enviados a clientes remotos, actualizados y enviados de vuelta al servidor.

Para su instalación sólo habrá que añadir una cadena al final de la variable del sistema CLASSPATH a un fichero que acompaña al Jakarta-Tomcat y que contiene el API JDBC 2.0:

```
<path Jakarta Tomcat>\common\lib\jdbc2_0-stdext.jar
```

## 7.9 MS Access

Esta instalación resulta muy sencilla, ya que viene incluida en el paquete Microsoft Office y que posee un muy buen asistente que nos guiará en su instalación.

A pesar de todo, tendremos que crear un puente entre nuestra aplicación y la base de datos, para lo cual deberemos asignar un Data Source Name (DSN). Este es un origen de datos de sistema **ODBC** en donde se almacena información de cómo debe conectarse a una base de datos.

De esta forma acudiremos a:

**Inicio > Panel de Control > Herramientas Administrativas >  
Orígenes de Datos (ODBC)**

En el menú de pestañas, seleccionaremos DSN de sistema e indicaremos que deseamos agregar un nuevo origen de datos al sistema. Seleccionaremos el controlador Driver de Microsoft Access (\*.mdb) y finalizaremos la operación.

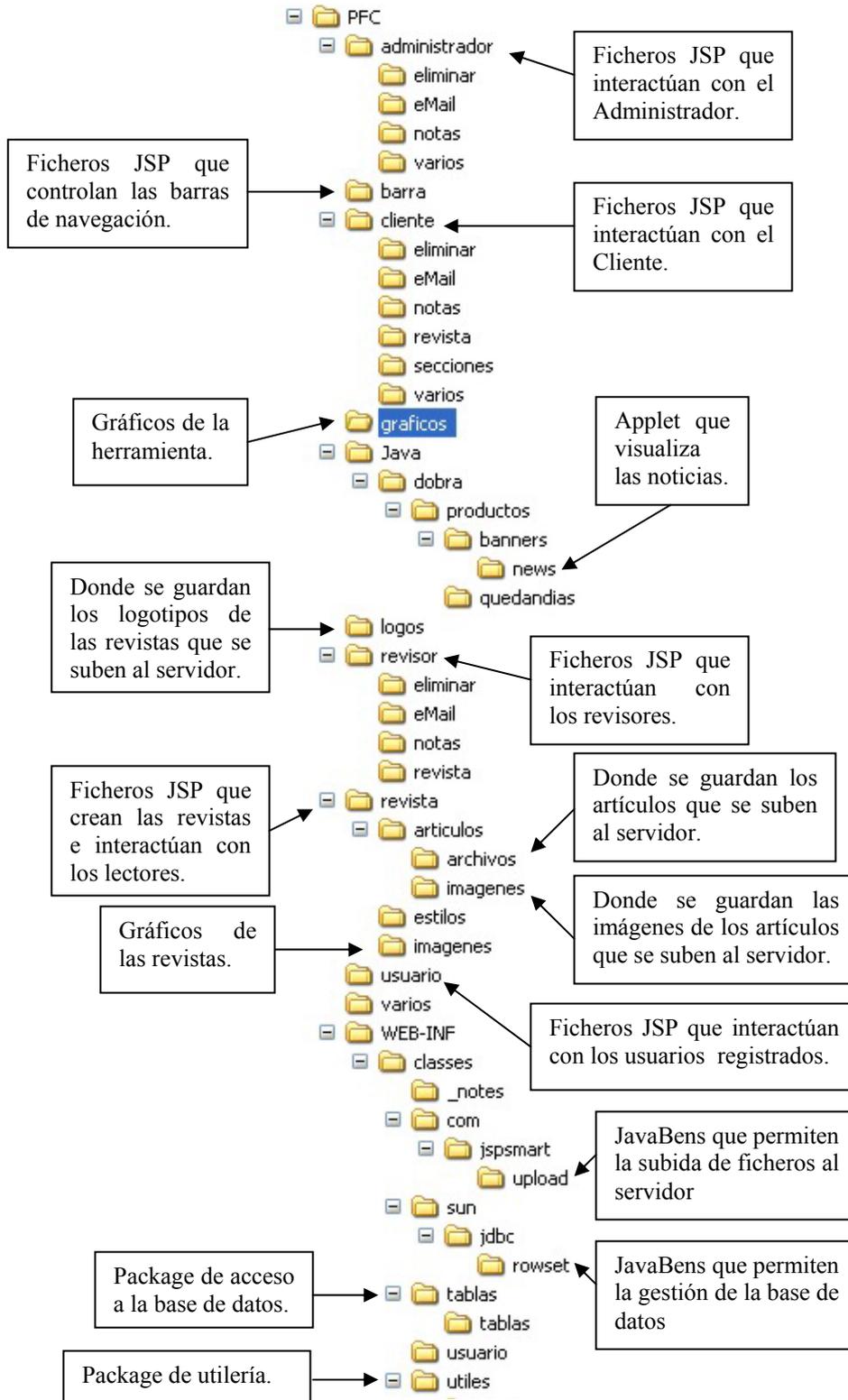
A continuación será solicitado el nombre del origen de datos (como queremos denominarlo) y se seleccionará la ruta a la base de datos que queramos emplear. Una vez hecho esto se aceptará la operación y tendremos creado el origen de datos que nos permitirá conectar la aplicación. Nuestra aplicación reconoce a la base de datos con el nombre de origen BaseDeDatos.

Para crear la base de datos en **MS Access** tan sólo es necesario ir construyendo las tablas en la vista de diseño de la forma fácil e intuitiva que esta herramienta proporciona.

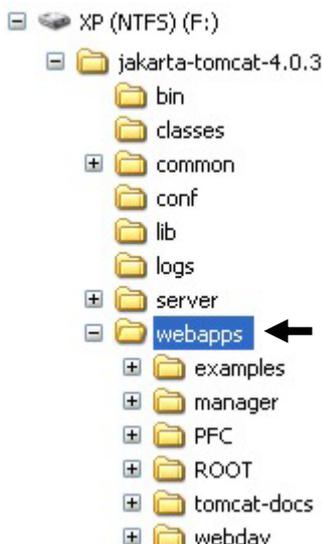
### 7.10 Instalación de nuestra herramienta

Una vez instalado los servidores y la base de datos, la instalación de nuestra herramienta es tan sencilla como la copia de una estructura de directorios.

La estructura de directorios de nuestra herramienta es la siguiente:



Ahora sólo deberemos de copiar toda esta estructura a la carpeta **Webapps** del Jakarta-Tomcat. En el siguiente gráfico se indica donde se ha de copiar.



Para acabar la instalación sólo nos queda introducir una entrada en el fichero de configuración del Jakarta-Tomcat llamado **server.xml** y que se encuentra en el directorio conf del Jakarta-Tomcat.

La línea a añadir será la siguiente y deberemos incluirla donde empiecen el resto de etiquetas de contexto:

```
<Context path="" docBase="PFC" debug="0"/>
```

Una vez finalizado todo el proceso de instalación de las herramientas y aplicaciones necesarias, la forma de poner en marcha nuestra herramienta será arrancando primero el Apache y después el Tomcat. De esta forma ya podremos conectarnos y empezar a trabajar con nuestra herramienta.

## 7.11 Bibliografía

[1] JavaServer Pages – Manual de usuario y tutorial  
Agustín Froufe  
Ra-Ma

[2] Aprenda Servlets de Java como si estuviera en segundo  
Javier García de jalón, José Ignacio Rodríguez y Aitor Imaz  
Escuela Superior de Ingenieros Industriales  
Universidad de Navarra

[3] Estrategias en la construcción de servidores de información  
Cuaderno del curso de extensión Universitaria de Oviedo  
8 de abril al 10 de mayo de 2002